

BBBBBBBBBBBBBB		AAAAAAAAAA	CCCCCCCCCCCC	KKK	KKK	UUU	UUU	PPPPPPPPPPPP
BBBBBBBBBBBBBB		AAAAAAAAAA	CCCCCCCCCCCC	KKK	KKK	UUU	UUU	PPPPPPPPPPPP
BBBBBBBBBBBBBB		AAAAAAAAAA	CCCCCCCCCCCC	KKK	KKK	UUU	UUU	PPPPPPPPPPPP
BBB	BBB	AAA	AAA	CCC	KKK	UUU	UUU	PPP
BBB	BBB	AAA	AAA	CCC	KKK	UUU	UUU	PPP
BBB	BBB	AAA	AAA	CCC	KKK	UUU	UUU	PPP
BBB	BBB	AAA	AAA	CCC	KKK	UUU	UUU	PPP
BBB	BBB	AAA	AAA	CCC	KKK	UUU	UUU	PPP
BBB	BBB	AAA	AAA	CCC	KKK	UUU	UUU	PPP
BBBBBBBBBBBBBB		AAA	AAA	CCC	KKKKKKKKKK	UUU	UUU	PPPPPPPPPPPP
BBBBBBBBBBBBBB		AAA	AAA	CCC	KKKKKKKKKK	UUU	UUU	PPPPPPPPPPPP
BBBBBBBBBBBBBB		AAA	AAA	CCC	KKKKKKKKKK	UUU	UUU	PPPPPPPPPPPP
BBB	BBB	AAAAAAAAAAAAAAAA	CCC	KKK	KKK	UUU	UUU	PPP
BBB	BBB	AAAAAAAAAAAAAAAA	CCC	KKK	KKK	UUU	UUU	PPP
BBB	BBB	AAAAAAAAAAAAAAAA	CCC	KKK	KKK	UUU	UUU	PPP
BBB	BBB	AAA	AAA	CCC	KKK	UUU	UUU	PPP
BBB	BBB	AAA	AAA	CCC	KKK	UUU	UUU	PPP
BBB	BBB	AAA	AAA	CCC	KKK	UUU	UUU	PPP
BBB	BBB	AAA	AAA	CCC	KKK	UUU	UUU	PPP
BBBBBBBBBBBBBB		AAA	AAA	CCCCCCCCCCCC	KKK	KKK	UUUUUUUUUUUUUUUU	PPP
BBBBBBBBBBBBBB		AAA	AAA	CCCCCCCCCCCC	KKK	KKK	UUUUUUUUUUUUUUUU	PPP
BBBBBBBBBBBBBB		AAA	AAA	CCCCCCCCCCCC	KKK	KKK	UUUUUUUUUUUUUUUU	PPP

RESTARTM
Table of contents

Reel checkpoint and restart

J 3

15-SEP-1984 23:37:37 VAX/VMS Macro V04-00

Page 0

(2) 59
(3) 128

CHECKPOINT_M
RESTART_M

Checkpoint machine state
Restart given saved machine state

```

0000 1      .TITLE RESTARTM      Reel checkpoint and restart
0000 2      .IDENT 'V04-000'
0000 3      :---
0000 4      :
0000 5      :*****
0000 6      :
0000 7      :*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 8      :*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 9      :*  ALL RIGHTS RESERVED.
0000 10     :
0000 11     :*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 12     :*  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 13     :*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 14     :*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 15     :*  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 16     :*  TRANSFERRED.
0000 17     :
0000 18     :*  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 19     :*  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 20     :*  CORPORATION.
0000 21     :
0000 22     :*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 23     :*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 24     :
0000 25     :
0000 26     :*****
0000 27     :
0000 28     :++
0000 29     : FACILITY:
0000 30     : Backup/Restore
0000 31     :
0000 32     : ABSTRACT:
0000 33     : This module contains routines to save and restore the low-level machine
0000 34     : state for reel checkpoint and restart.
0000 35     :
0000 36     : ENVIRONMENT:
0000 37     : VAX/VMS user mode.
0000 38     :
0000 39     :
0000 40     :--
0000 41     :
0000 42     : AUTHOR: M. Jack, CREATION DATE: 30-May-1981
0000 43     :
0000 44     :
0000 45     : MODIFIED BY:
0000 46     :
0000 47     : V03-001 MLJ0090      Martin L. Jack, 7-May-1982 13:26
0000 48     : Repair reel restart.
0000 49     :
0000 50     : V02-001 MLJ0054      Martin L. Jack, 22-Nov-1981 22:24
0000 51     : Integrate GET_VM and FREE_VM jacket routines.
0000 52     :
0000 53     :
0000 54     :**
0000 55     :
0000 56     :
00000000 57     .PSECT CODE,EXE,NOWRT

```



```

0000 59      .SBTTL CHECKPOINT_M          Checkpoint machine state
0000 60
0000 61      :++
0000 62      :
0000 63      : Functional Description:
0000 64      : This routine checkpoints the low-level machine state so that we can
0000 65      : later restart at the same point.
0000 66      :
0000 67      : Calling Sequence:
0000 68      : CALLS/CALLG
0000 69      :
0000 70      : Input Parameters:
0000 71      : 04(AP) = CHKPT_HIGH_SP
0000 72      : 08(AP) = Address of CHKPT_STACK
0000 73      : 12(AP) = Address of CHKPT_LOW_SP
0000 74      :
0000 75      : Implicit Inputs:
0000 76      : none
0000 77      :
0000 78      : Output Parameters:
0000 79      : none
0000 80      :
0000 81      : Implicit Outputs:
0000 82      : none
0000 83      :
0000 84      : Routines Called:
0000 85      : GET_VM
0000 86      :
0000 87      : Routine Value:
0000 88      : none
0000 89      :
0000 90      : Signals:
0000 91      : none
0000 92      :
0000 93      : Side Effects:
0000 94      : none
0000 95      :
0000 96      :--
0000 97
OFFC 0000 98      .ENTRY CHECKPOINT_M,*M<R2,R3,R4,R5,R6,R7,R8,R9,R10,R11>
0002 99      :
0002 100     : The entry mask specifies all registers so that when the RET in
0002 101     : RESTART_M executes, all registers are restored.
0002 102     :
0002 103     : Save the frame pointer in the global variable CHKPT_LOW_SP.
0002 104     :
OC BC 5D D0 0002 105     MOVL    FP,@12(AP)          ; Save stack base pointer
0006 106     :
0006 107     : Compute the difference between the saved frame pointer in the global
0006 108     : variable CHKPT_HIGH_SP and CHKPT_LOW_SP. This value is the length
0006 109     : of the section of stack that will be saved. It includes the stack
0006 110     : frame for this routine.
0006 111     :
56 04 AC 7D 0006 112     MOVQ    4(AP),R6          ; R6 = CHKPT_HIGH_SP
000A 113     : R7 = Address of CHKPT_STACK
7E 56 5D C3 000A 114     SUBL3    FP,R6,-(SP)      ; Compute size of stack to save
000E 115     :

```

RESTARTM
V04-000

Reel checkpoint and restart
CHECKPOINT_M Checkpoint machine state

M 3

15-SEP-1984 23:37:37 VAX/VMS Macro V04-00
4-SEP-1984 22:59:30 [BACKUP.SRC]RESTARTM.MAR;1

Page 3
(2)

			000E	116	:	Allocate the dynamic memory, placing the address of the allocated
			000E	117	:	space in the global variable CHKPT_STACK.
			000E	118	:	
00000000'GF	6E	DD	000E	119	:	PUSHL (SP) ; Copy size of stack to save
67	01	FB	0010	120	:	CALLS #1,G^GET_VM ; Allocate space to hold saved stack
	50	D0	0017	121	:	MOVL R0,(R7) ; Store address in CHKPT_STACK
			001A	122	:	
			001A	123	:	Copy the stack area to the dynamic memory, and return.
			001A	124	:	
00 B7	6D	6E	28	001A	:	MOVC3 (SP),(FP),a(R7) ; Save the stack area
			04	001F	:	RET
				126	:	


```
0020 128 .SBTTL RESTART_M Restart given saved machine state
0020 129
0020 130 :++
0020 131 :
0020 132 Functional Description:
0020 133 This routine restarts using the low-level machine state that was saved
0020 134 by CHECKPOINT_M.
0020 135 :
0020 136 Calling Sequence:
0020 137 CALLS/CALLG
0020 138 :
0020 139 Input Parameters:
0020 140 04(AP) = CHKPT_LOW_SP
0020 141 08(AP) = CHKPT_HIGH_SP
0020 142 12(AP) = Address of CHKPT_STACK
0020 143 :
0020 144 Implicit Inputs:
0020 145 none
0020 146 :
0020 147 Output Parameters:
0020 148 none
0020 149 :
0020 150 Implicit Outputs:
0020 151 none
0020 152 :
0020 153 Routines Called:
0020 154 none
0020 155 :
0020 156 Routine Value:
0020 157 none
0020 158 :
0020 159 Signals:
0020 160 none
0020 161 :
0020 162 Side Effects:
0020 163 none
0020 164 :
0020 165 :--
0000 0020 166 .ENTRY RESTART_M,^M<> ; Register save irrelevant
0022 168 :
0022 169 Compute the difference between the saved frame pointer in the global
0022 170 variable CHKPT_HIGH_SP and CHKPT_LOW_SP. This value is the length
0022 171 of the section of stack that will be restored from the saved copy.
0022 172 :
56 08 AC 7D 0022 173 MOVQ 8(AP),R6 ; R6 = CHKPT_HIGH_SP
0026 174 ; R7 = Address of CHKPT_STACK
56 04 AC C2 0026 175 SUBL2 4(AP),R6 ; Compute length of saved area
002A 176 :
002A 177 Now restore the FP and SP registers to the values they had on entry to
002A 178 routine CHECKPOINT_M. Then, copy the saved section of stack to the
002A 179 stack. This restores the stack to the exact state that it was
002A 180 in on entry to routine CHECKPOINT_M. It is important not to cause
002A 181 any signals or other stack activity during these three instructions.
002A 182 :
5D 04 AC D0 002A 183 MOVL 4(AP),FP ; Restore stack pointers
5E 5D D0 002E 184 MOVL FP,SP ;
```

RESTARTM
V04-000

Reel checkpoint and restart

RESTART_M Restart given saved machine s

B 4

15-SEP-1984 23:37:37

VAX/VMS Macro V04-00

Page

5

4-SEP-1984 22:59:30

[BACKUP.SRC]RESTARTM.MAR;1

(3)

6D 00 B7 56 28 0031 185
0036 186 :
0036 187 :
0036 188 :
0036 189 :
0036 190 :
0036 191 :
04 0036 192

MOV C3 R6,a(R7),(FP)

; Restore stack

The following RET instruction executes in the context of routine CHECKPOINT_M, because the stack has been restored, and returns to the caller of routine CHECKPOINT_M. The previous execution flow has been entirely lost.

RET

; Return to caller of CHECKPOINT_M

RESTARTM
V04-000

C 4
Reel checkpoint and restart 15-SEP-1984 23:37:37 VAX/VMS Macro V04-00
RESTART_M Restart given saved machine s 4-SEP-1984 22:59:30 [BACKUP.SRC]RESTARTM.MAR;1 Page 6
0037 194 .END (4)

RESTARTM
Symbol table

Reel checkpoint and restart

D 4

15-SEP-1984 23:37:37 VAX/VMS Macro V04-00
4-SEP-1984 22:59:30 [BACKUP.SRC]RESTARTM.MAR;1

Page 7
(4)

CHECKPOINT_M 00000000 RG 01
GET_VM ***** X 01
RESTART_M 00000020 RG 01

+-----+
! Psect synopsis !
+-----+

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
CODE	00000037 (55.)	01 (1.)	NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC BYTE

+-----+
! Performance indicators !
+-----+

Phase	Page faults	CPU Time	Elapsed Time
Initialization	13	00:00:00.08	00:00:00.86
Command processing	88	00:00:00.62	00:00:03.16
Pass 1	72	00:00:00.44	00:00:02.14
Symbol table sort	0	00:00:00.00	00:00:00.00
Pass 2	52	00:00:00.33	00:00:01.34
Symbol table output	2	00:00:00.01	00:00:00.01
Psect synopsis output	1	00:00:00.02	00:00:00.09
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	230	00:00:01.50	00:00:07.60

The working set limit was 750 pages.
2036 bytes (4 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 3 non-local and 0 local symbols.
194 source lines were read in Pass 1, producing 17 object records in Pass 2.
0 pages of virtual memory were used to define 0 macros.

+-----+
! Macro library statistics !
+-----+

Macro library name	Macros defined
_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:RESTARTM/OBJ=OBJ\$:RESTARTM MSRC\$:RESTARTM/UPDATE=(ENH\$:RESTARTM)

0013 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

